

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of)
)
MARTIN THOMAS MASON)
DAVID ANDREW MC CONNELL)
and)
AJITHKUMAR VENKATA DASARI)
)
Serial No. NOT YET ASSIGNED) Group Art Unit:
)
Filed: February 13, 2001) Examiner:
)
For: SOFTWARE TOOL TO ALLOW FIELD)
PROGRAMMABLE SYSTEM LEVEL)
DEVICES)
)

Honorable Commissioner of Patents
and Trademarks
Washington, D.C. 20231

Sir:

PRELIMINARY AMENDMENT A

Prior to the initial Office Action, amend the application
as follows:

IN THE DRAWINGS:

Amend FIG. 3, as shown in red in the attached drawing.

IN THE SPECIFICATION:

Page 3, line 11, change "mostly" to --most--;
Page 6, line 8, delete "March 23, 1999-Xilinx Inc.";
line 22, change "cots" to --cost--;

Page 11, line 13, after "simulation" insert --, programmed
 logic,--;
 line 14, after "simulation" insert --, program
 code,--;
 lines 15-17 delete the sentence starting with "The" and
 ending with "FPGA device.";

line 20, change "excample" to --example--;

Page 12, line 3, before "memory" insert --FPSLIC--;
 line 6, delete "16K X";
 line 7, delete "16 or";
 line 8, after "were required" insert --for
 instruction storage--;
 line 9, delete "program";
 line 10, delete "eight 4 kilobyte";
 line 12, delete "eight four kilobyte, or";

Page 13, line 7, after "hardware" insert --(FPGA)--;
 line 8, after "software" insert --(micro-
 controller)--;

Page 14, line 1, change "32" to --36--;

Page 15, line 3, delete "16K X 16 or";
 line 6, after "kilobyte blocks" insert --, by way of
 example--;
 line 8, delete "eight four kilobyte, or";

Page 16, line 4, change "A 30 MIPS" to --The AVR--;

Page 17, line 18, after "hardware" insert --(logic)-- and
 after "software" insert --(program code)--;

line 25, delete the entire line starting with "An
TA94K0";

Page 18, lines 1-7, delete lines 1-7 in their entirety;

Page 19, line 25, delete "achieve 30+ MIPS";

Page 20, line 1, delete "throughput, allowing to";

line 24, at end of the line, insert --.--;

line 25, delete the entire line, starting with
"additional";

Page 21, line 1, delete "in 1999.";

lines 2-25, delete lines 2-25, effectively deleting the
entire page;

Page 22, lines 1-6 delete lines 1-6;

delete the entire paragraph starting at line
22 with "Design tools" through page 23, line
10;

Page 25, line 9, change "7.0" to --tools--;

Page 29, line 8, change "412" to --42--;

line 12, change "simulator port" to --simulation
post--;

lines 13-14, change "simulator port" to --simulation
post--;

line 17, change "simulator port" to --simulation
post--; and

Page 30, line 16, change "simulator port" to --simulation
post--.

IN THE CLAIMS:

1. (Once Amended) A method for designing a field programmable system level integrated circuit (FPSLIC) [co-verifying a hardware simulation of a field-programmable-gate array (FPGA) and a software simulation of the field-programmable-gate array], comprising the steps of:

simulating [in hardware a FPGA device] , pre-layout, the field programmable system level integrated circuit using co-verification software;

generating, [from] a [the] simulation [in hardware, a simulator -port] post layout model of the implementation for co-verification of the field programmable system level integrated circuit [FPGA device];

simulating[, with an instruction-set simulator,] program code; [in software the FPGA device;]

outputting register contents from the instruction-set software, [from the simulation in software] for the co-verification of the field programmable system level integrated circuit; [and

verifying contents from the simulator-port layout with the register contents]

creating a bit stream for a field programmable gate array (FPGA) device part of the field programmable system level integrated circuit;

creating a program code for a micro-controller part of the field programmable system level integrated circuit; and

creating, from the bit stream and the program code, in a program file, a combined-bit stream that has data for both the FPGA device and instruction code for a RISC micro-controller.

2. (Once Amended) The method as set forth in claim 1, further including the steps of:

outputting peripheral contents from [the] instruction-set [simulator, from the simulation in] software; and

verifying contents from the [simulator-port] simulation post layout with the peripheral contents.

3. (Once Amended) The method as set forth in claim 1, further including the steps of:

outputting UART contents from the instruction-set [simulator, from the simulation in software] software; and

verifying contents from the [simulator-port] simulation post layout with the UART contents.

4. (Once Amended) A system for designing a field programmable system level integrated circuit [co-verifying a hardware simulation of a field-programmable-gate array (FPGA) and a software simulation of the field-programmable-gate array], comprising:

a [hardware] software simulator for simulating, pre-layout, a field programmable system level integrated circuit using co-verification software [a FPGA device, with the hardware

simulator having a simulator-port layout of the FPGA device] ;

instruction set software for outputting register contents for the co-verification of the field programmable system level integrated circuit;

5 [a software simulator for simulating the FPGA device, with software simulator having an instruction-set simulator for outputting register contents; and

verification software for verifying contents from the simulator-port layout with the register contents]

10 field programmable gate array (FPGA) software for creating a bit stream for an FPGA device part of the FPSLIC, and for creating a program code for the micro-controller part of the FPSLIC; and

15 said FPGA software for creating, from the bit stream and the program code a combined-bit stream that has data for both the FPGA device and the instruction code for a RISC micro-controller.

5. (Once Amended) The system as set forth in claim 4, with:

said instruction-set [simulator] software for outputting peripheral contents; and

said verification software for verifying contents from the [simulator-port] simulation post layout with the peripheral contents.

6. (Once Amended) The system as set forth in claim 4, with:

said instruction-set simulator] software for
outputting UART contents; and

said verification software for verifying contents from
the [simulator-port] simulation post layout with the UART
contents.

Add the following claims:

--7. A method for designing a field programmable system
level integrated circuit (FPSLIC), comprising the steps of:

simulating, pre-layout, the field programmable system
level integrated circuit using co-verification software;

generating, post layout, a simulation model of an
implementation for co-verification of the field programmable
system level integrated circuit;

simulating, with an instruction-set simulator, program
code;

outputting register contents from the instruction-set
simulator, for the co-verification of the field programmable
system level integrated circuit;

creating a bit stream for a field programmable gate
array (FPGA) device part of the field programmable system level
integrated circuit; and

creating a program code for a micro-controller part of
the field programmable system level integrated circuit.

8. The method as set forth in claim 7, further including the steps of:

outputting peripheral contents from instruction-set software; and

verifying contents from the simulation post layout with the peripheral contents.

9. The method as set forth in claim 7, further including the steps of:

outputting UART contents from the instruction-set software; and

verifying contents from the simulation post layout with the UART contents.

10. A system for designing a field programmable system level integrated circuit, comprising:

a software simulator for simulating, pre-layout, a field programmable system level integrated circuit using co-verification software;

an instruction-set simulator for simulating program code and for outputting register contents for co-verification of the field programmable system level integrated circuit; and

field programmable gate array (FPGA) software for creating a bit stream for an FPGA device part of the FPLSIC, and for creating a program code for the micro-controller part of

the FPSLIC.

11. The system as set forth in claim 10, with:
said instruction-set simulator for outputting
peripheral contents; and
said verification software for verifying contents from
the simulation post layout with the peripheral contents.

12. The system as set forth in claim 10, with:
said instruction-set simulator for outputting UART
contents; and
said verification software computer for verifying
contents from the simulation post layout with the UART contents.

13. A system for designing a field programmable system
level integrated circuit, comprising:

software means for simulating, pre-layout, a field
programmable system level integrated circuit using co-
verification software;

instruction means for simulating program code and for
outputting register contents for co-verification of the field
programmable system level integrated circuit; and

field programmable gate array (FPGA) means for
creating a bit stream for an FPGA device part of the FPSLIC, and
for creating a program code for the micro-controller part of
the FPSLIC.

14. The system as set forth in claim 13, with:
said instruction means for outputting peripheral contents; and
verification software for verifying contents from the simulation post layout with the peripheral contents.

15. The system as set forth in claim 13, with:
said instruction means for outputting UART contents;
and
verification software for verifying contents from the simulation post layout with the UART contents.

16. A system for designing a field programmable system level integrated circuit (FPSLIC), comprising:

software means for simulating, pre-layout, a field programmable system level integrated circuit using co-verification software;

instruction set software for simulating program code and for outputting register contents for co-verification of the field programmable system level integrated circuit;

field programmable gate array (FPGA) means for creating a bit stream for an FPGA device part, and for creating a program code for the micro-controller part of the FPSLIC; and

said FPGA means for creating, from the bit stream and the program code, in a program file, a combined-bit stream that

has data for both the FPGA device and the instruction code for a RISC micro-controller.

17. The system as set forth in claim 16, with:
said instruction-set software for outputting peripheral contents; and
software means for verifying contents from the simulation post layout with the peripheral contents.

18. The system as set forth in claim 16, with:
said instruction-set software for outputting UART contents; and
said verification software for verifying contents from the simulation post layout with the UART contents.--

REMARKS

By this amendment, applicant amends claims 1-6, and adds claim 7-18. Claims 1-18 are pending in the application.

Respectfully submitted,

DAVID NEWMAN CHARTERED

By: 

Date: February 12, 2001

David B. Newman, Jr.
Registration No. 30,966

T:\ATML\ATML04US.PAM.wpd

LAW OFFICES
DAVID NEWMAN
CHARTERED
ANTENNIAL SQUARE
P.O. BOX 2728
PLATA, MD 20646
(301) 934-6100